

1. Cluster -

Clusters (1) Info								
<input type="text" value="Filter clusters"/>								
	Cluster name	Status	Kubernetes version	Support period	Upgrade policy	Created	Provider	
<input type="radio"/>	nodejs-app-cluster	Active	1.29 Upgrade now	Extended support until March 23, 2026	Extended	an hour ago	EKS	

2. Spot instance -

<

3. Spot instance on cli as k8s node -

```
$ sudo kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-10-0-2-183.ap-south-1.compute.internal Ready    <none>   18m    v1.29.15-eks-113cf36
$
```

4. Deploy mongodb -

```
$ sudo kubectl apply -f mongo-secrets.yaml,mongodb-deployment.yaml
secret/mongodb-secret created
storageclass.storage.k8s.io/ebs-sc created
service/mongodb-service created
statefulset.apps/mongodb created
$ sudo kubectl get pvc -l app=mongodb
NAME                                STATUS    VOLUME                                     CAPACITY  ACCESS MODES  STORAGECLASS  VOLUMEATTRIBUTESCLASS  AGE
mongodb-storage-mongodb-0           Bound    pvc-3840c369-7a0d-43e4-a6cb-f9da3041d57c  1Gi        RWO            ebs-sc        <unset>                 12s
$ sudo kubectl get storageclass
NAME      PROVISIONER      RECLAIMPOLICY  VOLUMEBINDINGMODE  ALLOWVOLUMEEXPANSION  AGE
ebs-sc    ebs.csi.aws.com  Delete        WaitForFirstConsumer  false                 22s
gp2 (default)  kubernetes.io/aws-ebs  Delete        WaitForFirstConsumer  false                 71m
$ sudo kubectl get statefulsets
NAME      READY  AGE
mongodb  1/1    30s
$
```

5. Deploy nodejs -

```
$ sudo kubectl apply -f nodejs-deployment.yaml
service/nodejs-service created
deployment.apps/nodejs-app-deployment created
$ sudo kubectl get pods
NAME                                READY   STATUS             RESTARTS   AGE
mongodb-0                          1/1     Running            0          80s
nodejs-app-deployment-56c95c79b4-7xxx9 0/1     ContainerCreating  0          6s
$ sudo kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
mongodb-0                          1/1     Running   0          102s
nodejs-app-deployment-56c95c79b4-7xxx9 1/1     Running   0          28s
$ _
```

6. Deploy alb -

```
$ sudo kubectl apply -f ingress.yaml
Warning: annotation "kubernetes.io/ingress.class" is deprecated, please use 'spec.ingressClassName' instead
ingress.networking.k8s.io/nodejs-ingress created
$ _
```

The screenshot shows the Amazon Elastic Kubernetes Service (EKS) console for a cluster named 'nodejs-app-cluster'. The left sidebar contains navigation links for Dashboard, Clusters, Settings, Amazon EKS Anywhere, and Related services. The main content area is divided into two sections: 'Resource types' and 'Workloads: Pods (11)'. The 'Resource types' section lists various Kubernetes resources like PodTemplates, ReplicaSets, Deployments, etc. The 'Workloads: Pods (11)' section shows a table of 11 pods, all in a 'Running' state. The pods include two load balancer controllers, two coreDNS pods, two CSI controllers, one kube-proxy, and one mongodb-0 pod.

Name	Created	Status
aws-load-balancer-controller-9647f8574-4nlg4	14 minutes ago	Running
aws-load-balancer-controller-9647f8574-v76kd	14 minutes ago	Running
aws-node-kp7x4	an hour ago	Running
coredns-5d5f56f475-bjttl	2 hours ago	Running
coredns-5d5f56f475-kh95j	2 hours ago	Running
ebs-csi-controller-559555cb49-8xj84	an hour ago	Running
ebs-csi-controller-559555cb49-zrxq9	an hour ago	Running
ebs-csi-node-9dz8v	an hour ago	Running
kube-proxy-c42s9	an hour ago	Running
mongodb-0	35 minutes ago	Running

```
$ sudo kubectl describe ingress nodejs-ingress
Name:          nodejs-ingress
Labels:        <none>
Namespace:     default
Address:       k8s-default-nodejsin-33ce08d7a8-853426368.ap-south-1.elb.amazonaws.com
Ingress Class: <none>
Default backend: <default>
Rules:
  Host      Path  Backends
  ---      -
  *         /    nodejs-service:80 (10.0.2.91:3000)
Annotations: alb.ingress.kubernetes.io/scheme: internet-facing
              alb.ingress.kubernetes.io/target-type: ip
              kubernetes.io/ingress.class: alb
Events:
  Type      Reason              Age   From      Message
  ---      -
  Normal    SuccessfullyReconciled 8m16s ingress Successfully reconciled
$ sudo kubectl get ingress nodejs-ingress
NAME      CLASS  HOSTS                                ADDRESS                                PORTS   AGE
nodejs-ingress <none> *      k8s-default-nodejsin-33ce08d7a8-853426368.ap-south-1.elb.amazonaws.com 80      8m25s
$
```

8. Access the endpoint -

The screenshot shows a web browser with the address bar displaying `k8s-default-nodejsin-33ce08d7a8-853426368.ap-south-1.elb.amazonaws.com/health`. The page content is a simple JSON response: `{"status": "ok", "db": "connected"}`. Below this, there is a screenshot of a web application titled "TODO LIST". The application has a text input field labeled "Enter task..", an "Add Todo" button, and a list containing the word "done". At the bottom of the application, there is a link to "Connect with me on LinkedIn" and a copyright notice: "© 2025 Ritik Gupta. All rights reserved."

9. Datadog alert -

The screenshot shows the Datadog Monitors interface. The top navigation bar includes the Datadog logo, a search bar, and a sidebar with various monitoring tools. The main content area displays a monitor titled "[Kubernetes] Pod {{pod_name.name}} is ImagePullBackOff on namespace {{kube_namespace.name}}". The monitor is in an "ALERT" state, indicated by a red "ALERT" badge. The monitor configuration shows it is a "Metric Monitor" created on Oct 17, 2025, at 5:01 am, with the integration set to "kubernetes" and the monitor pack set to "kubernetes". The monitor behavior is defined by a query: `max(last_10m):default_zero(max:kubernetes_state.container.status_report..`. The monitor is currently "Muted". The alert timeline shows a single alert event on Oct 17, 5:00:31 am, with the message: "pod {{pod_name.name}} is in ImagePullBackOff on {{kube_namespace.name}} This could happen for several reasons, for example a bad image path or tag or if the credentials for pulling images are not configured properly." The alert has 2 recipients. The message template is displayed below the timeline. The interface also includes a "Next Steps" section with a "Mute" button and a "Declare Incident" button.

10. Delete resources -

```
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=nodejs-app-cluster, 02m50s elapsed]
module.eks.aws_eks_cluster.this[0]: Destruction complete after 2m54s
module.eks.aws_iam_role_policy_attachment.this["AmazonEKSVPCResourceController"]: Destroying... [id=nodejs-app-cluster-cluster-20251017042640929100000002-20251017042643496900000006]
module.eks.module.kms.aws_kms_key.this[0]: Destroying... [id=005d5774-eaf4-4aa2-928f-b6c3bbd733b5]
module.eks.aws_security_group_rule.node["ingress_cluster_8443_webhook"]: Destroying... [id=sgrule-192200564]
module.eks.aws_cloudwatch_log_group.this[0]: Destroying... [id=/aws/eks/nodejs-app-cluster/cluster]
module.eks.aws_security_group_rule.node["ingress_cluster_6443_webhook"]: Destroying... [id=sgrule-405388245]
module.eks.aws_security_group_rule.node["ingress_self_coredns_tcp"]: Destroying... [id=sgrule-4228507080]
module.eks.aws_security_group_rule.node["ingress_cluster_4443_webhook"]: Destroying... [id=sgrule-3513443523]
module.vpc.aws_subnet.private[0]: Destroying... [id=subnet-03dfcbafe845e87ba]
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destroying... [id=sgrule-2886904360]
module.vpc.aws_subnet.private[2]: Destroying... [id=subnet-0f9732777f93d697a]
module.eks.module.kms.aws_kms_key.this[0]: Destruction complete after 1s
module.eks.aws_cloudwatch_log_group.this[0]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["ingress_cluster_kubelet"]: Destroying... [id=sgrule-2503021465]
module.eks.aws_security_group_rule.node["ingress_cluster_443"]: Destroying... [id=sgrule-1134747375]
module.vpc.aws_subnet.private[2]: Destruction complete after 1s
module.vpc.aws_subnet.private[0]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["egress_all"]: Destroying... [id=sgrule-2839358688]
module.eks.aws_security_group_rule.node["ingress_self_coredns_udp"]: Destroying... [id=sgrule-1482049063]
module.eks.aws_security_group_rule.node["ingress_cluster_8443_webhook"]: Destruction complete after 1s
module.vpc.aws_subnet.private[1]: Destroying... [id=subnet-0a3f187ad2a6d24b8]
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["ingress_cluster_9443_webhook"]: Destroying... [id=sgrule-1876266175]
module.vpc.aws_subnet.private[1]: Destruction complete after 0s
module.eks.aws_security_group_rule.node["ingress_nodes_ephemeral"]: Destroying... [id=sgrule-2753739322]
module.eks.aws_iam_role_policy_attachment.this["AmazonEKSVPCResourceController"]: Destruction complete after 1s
module.eks.aws_iam_role_policy_attachment.this["AmazonEKSClusterPolicy"]: Destroying... [id=nodejs-app-cluster-cluster-20251017042640929100000002-20251017042643598500000007]
module.eks.aws_security_group_rule.node["ingress_cluster_4443_webhook"]: Destruction complete after 2s
module.eks.aws_iam_role_policy_attachment.this["AmazonEKSClusterPolicy"]: Destruction complete after 1s
module.eks.aws_iam_role.this[0]: Destroying... [id=nodejs-app-cluster-cluster-20251017042640929100000002]
module.eks.aws_security_group_rule.node["ingress_self_coredns_tcp"]: Destruction complete after 2s
module.eks.aws_security_group_rule.node["ingress_cluster_6443_webhook"]: Destruction complete after 2s
module.eks.aws_iam_role.this[0]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["ingress_cluster_kubelet"]: Destruction complete after 2s
module.eks.aws_security_group_rule.node["ingress_cluster_443"]: Destruction complete after 2s
module.eks.aws_security_group_rule.node["egress_all"]: Destruction complete after 3s
module.eks.aws_security_group_rule.node["ingress_self_coredns_udp"]: Destruction complete after 3s
module.eks.aws_security_group_rule.node["ingress_cluster_9443_webhook"]: Destruction complete after 4s
module.eks.aws_security_group_rule.node["ingress_nodes_ephemeral"]: Destruction complete after 5s
module.eks.aws_security_group.cluster[0]: Destroying... [id=sg-0302d4fb77a2f5b23]
module.eks.aws_security_group.node[0]: Destroying... [id=sg-0e5ab054b64d14f29]
module.eks.aws_security_group.cluster[0]: Destruction complete after 0s
module.eks.aws_security_group.node[0]: Destruction complete after 0s
module.vpc.aws_vpc.this[0]: Destroying... [id=vpc-09b0c30cb2e75bb29]
module.vpc.aws_vpc.this[0]: Destruction complete after 1s

Destroy complete! Resources: 66 destroyed.
```